

## Modellierung und Programmierung 1 Übungsserie 5

### Lösungsvorschläge

#### 1. OOP - Implementieren / Collection

##### a) *Klasse Ausstattung*

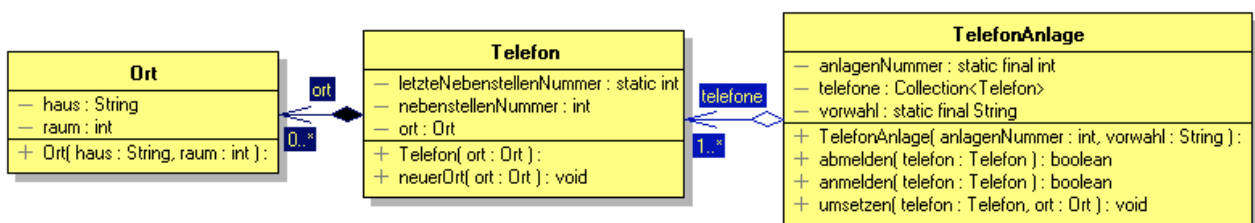
<b>Instanzvariablen</b>	automatik klima navi park sonst winter
<b>Instanzmethoden</b>	berechneGesamtzuschlag reifenWechsel setSonst toString
<b>Klassenvariablen</b>	zuschlag
<b>Klassenmethoden</b>	setZuschlag main

Der Zuschlag ist für alle Ausstattungen gleich, d.h. unabhängig vom Objekt. Die Klassenmethoden `setZuschlag` verändert den Wert der Klassenvariablen `zuschlag`. Die Methode `main` startet das Programm.

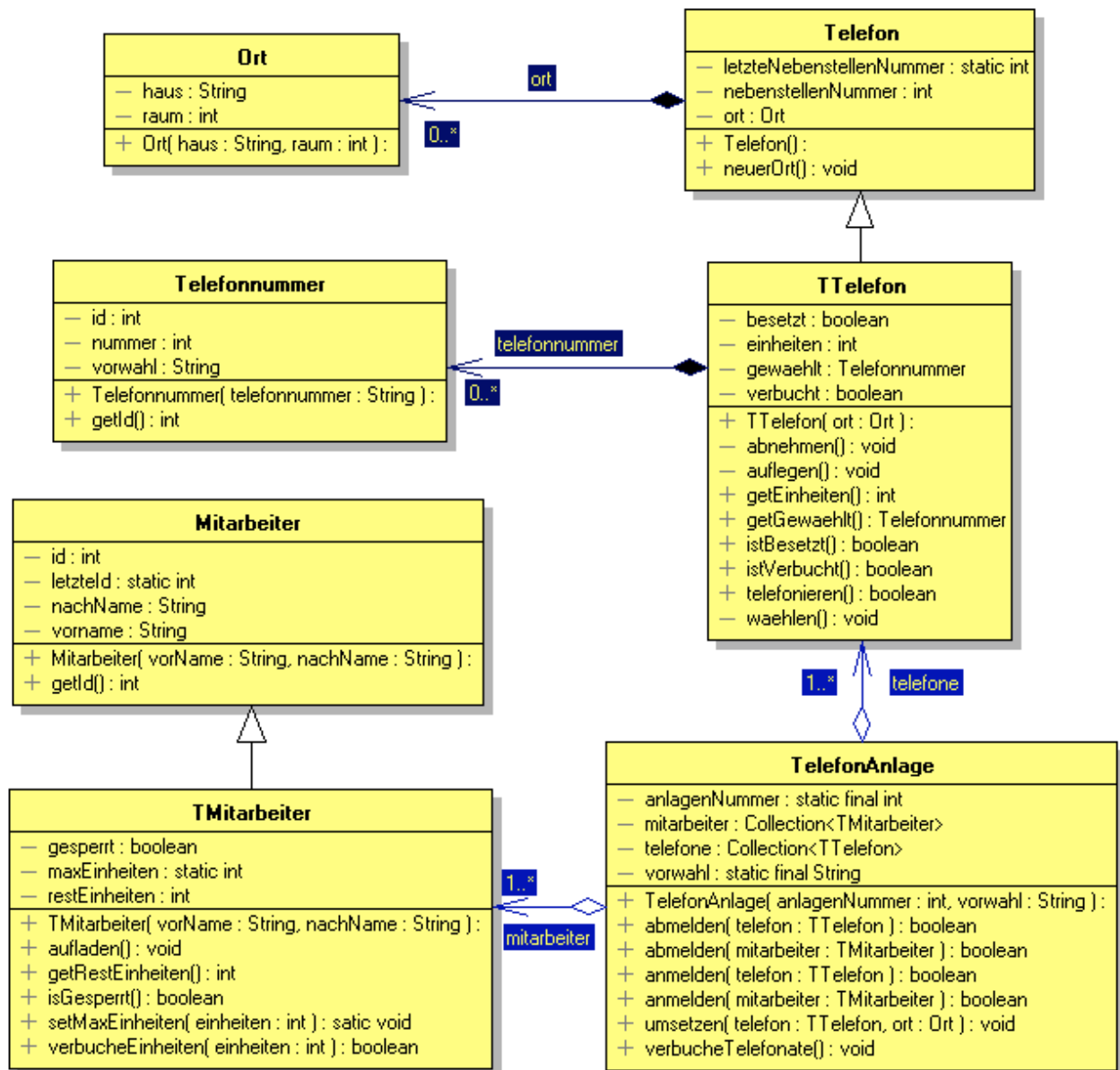
- b) siehe *Ausstattung.java* und siehe *Ausstattung.out*  
c) siehe *Auto.java*, *Kleinwagen.java*, *Mittelklassewagen.java*, *Oberklassewagen.java* und siehe *Auto.out*  
d) siehe *Ort.java*, *Parkplatz.java*, *Strasse.java*, *Werkstatt.java* und siehe *Ort.out*  
e) , f) und g) *Fuhrpark.java*

#### 2. OOP - Modellierung / Collection

##### a) *Telefonanlage*



b) *Mitarbeiter*



c) *Collection*

telefone: ArrayList<TTelefon> oder Vector<TTelefon>, wahlfreie Zugriffe überwiegen.  
 mitarbeiter: LinkedList<TMitarbeiter>, falls Mitarbeiter häufiger die Firma wechseln, sonst  
 ArrayList<TMitarbeiter> oder Vector<TMitarbeiter>, wahlfreie Zugriffe überwiegen.